SIIT '99
Proceedings

1st IEEE
Conference on
Standardisation
and Innovation
in Information
Technology

Aachen
September 15–17, 1999

Sponsored by

IEEE

RWTH

Standardisation and Innovation in Information Technology

SIIT '99

- Proceedings -

Aachen, Germany

September 15 - 17, 1999

Conference Record

Standardisation and Innovation in Information Technology
SIIT '99

Aachen, Germany

Edited by Kai Jakobs & Robin Williams

ISBN 0-7803-9935-8

Table of Contents

Paper Session 1: Organisational Aspects of Standardisation

Paper Session 2: The Standards Setting Process

Debate: Is There a Future for the Internet's standardisation process?

Poster Session: The Many Facets of Standardisation

Paper Session 3: On Innovation

Paper Session 4: Economic Issues and Theory of Standardisation

Paper Session 5: Specific Standards and Applications

Workshop: 'Electronic Tools to Support Standardisation Work'

Panel: Standardisation in the Next Millenium - Where Do We Go from Here?

Keynote and Invited Papers

Using The ISO/IEC 12207 Tailoring Process for Defining a Maintenance Process[1]

Macario Polo, Mario Piattini, Francisco Ruiz, Coral Calero
Escuela Superior de Informática
Universidad de Castilla-La Mancha
e-mail: {mpolo, mpiattini, fruiz, ccalero}@inf-cr.uclm.es

Abstract

ISO/IEC 12207 is a global framework applicable to all software life cycle processes: from the acquisition of a software product, service or system, to its maintenance, passing through the supply, development and operation. This standard is tailorable to different software projects. As ISO/IEC 12207 says, "the tailoring process is the deletion of non-applicable processes, activities and tasks".

On the other hand, maintenance is the most costly stage of software life cycle, representing sometimes 90% of total costs. Possibly, these high costs have a part of its origin in the lack of a methodology to manage maintenance efficiently. However, it is surprising that most the organizations asked about this, which follow a methodology for their new developments, affirm they have not any one for maintenance [6]. Therefore, we agree with [2] when they say that in order to accomplish the effective management of the software release process, methodologies that take into account the maintenance release process are needed. The large number of legacy systems still being maintained invites us to propose possible improvements for software maintenance.

Due to the big impact of ISO/IEC 12207 on software industry and organizations, we have applied its tailoring process to the rest of the standard for obtaining in this manner a methodology for software maintenance. This application is the purpose of this paper.

I. INTRODUCTION

ISO/IEC 12207 [5] is a global framework which covers all aspects of software life cycle processes. It is applicable to the Acquisition, Supply, Development, Operation and Maintenance of software products, systems and services. This standard is tailorable to different software projects: among these ones, maintenance projects.

Maintenance, as it is well known, is the most costly stage of the software life cycle. Some studies affirm that maintenance costs reach sometimes 90% of the total costs of the life cycle [4]. Although, these ones are habitual costs for most organizations, they do not follow any methodology to manage software maintenance, but they follow it out for new developments (whose total effort represents between 10% and 40%).

With these circumstances, it is not rare that the software industry demand methodologies for controlling and managing this hard and long stage of the software life cycle, above all if we take into account the big number of legacy systems still being maintained [3]. For [2], methodologies which take into account the maintenance release process are needed to accomplish the effective management of the software release process.

In this paper we present the application of the ISO/IEC 12207 Tailoring process to build-up a methodology for software maintenance. We have chosen this standard because, as [7] says, it "will drive the world software trade and will impact on maintenance". Also ISO/IEC 12207 it is the basis for the development of other standards, as Spice (ISO/IEC 15504).

This paper is organized as follows: section II is a revision of the Tailoring process of ISO/IEC 12207 and a brief description of the rest of processes, but paying special attention to Maintenance. In section III we describe the use of the Tailoring process to ISO/IEC 12207 to build a methodology for maintaining information systems. In section IV we state our conclusions and future work.

We wish that this methodology will be adequate to the standard, since "Compliance is defined as the performance of all the processes, activities and tasks selected from this International Standard in the Tailoring Process for the software project". If we could reach this goal, we would be ready to provide to organizations a useful methodology, especially for those ones that want to achieve certifications as, for example, ISO 9000.

[1] This work is partially supported by MÁNTICA (CICYT/European Union 1FD-097-0168) and MANTEMA projects (carried out by Atos ODS, S.A. and Universidad de Castilla-La Mancha; ATYCA, Dirección General de Tecnología y Seguridad Industrial of the MINER, Ministerio de Industria y Energía, Spain).

0-7803-9935-8/$10.00 © 1999 IEEE

II. THE TAILORING PROCESS AND ISO/IEC 12207

ISO/IEC 12207 defines a set of processes which must be performed during the software life cycle, independently of which one the organization had selected. Through its Tailoring process, this standard is tailorable for different software life cycle models, and it is the organizations responsibility to map the activities and tasks of the standard into the chosen model.

The standard divides the processes into three groups (plus the Tailoring process):

1) Primary processes: those which serve primary parties (acquirer, supplier, developer, operator and maintainer) during the software life cycle.
2) Supporting processes: those which support another process as an integral part with a distinct purpose, contributing to the successful performance of the project and to its quality.
3) Organizational processes: they are used by the organization for establishing and implementing of an underlying structure, which allows the continuous improvement of both the structure and processes.

A. The Tailoring process

In ISO/IEC 12207, every process is defined as a set of activities and, every activity, as a set of tasks. The Tailoring, of course, is also defined in such manner. In particular, the activities which compose Tailoring are the following:

1) Identifying project environment, whose only task consists of identifying those project environment characteristics which will influence on the tailoring (e.g., the life cycle model, system and software requirements, size, criticality and system types, software product or service, etc.).
2) Soliciting inputs, whose only task consists in the solicitation of the inputs of the organizations which will be affected by the tailoring decisions (e.g., users, support personnel, etc.).
3) Selecting processes, activities and tasks, which (without underestimating the importance of the others) is the main task of Tailoring. This activity consists of three tasks:
   3.1. Decision about what processes, activities and tasks will be performed, including the documentation to be developed and responsible.
   3.2. Provision of the processes, activities and tasks selected in the previous task and not provided by ISO/IEC 12207.
   3.3 Paying special attention to those processes, activities and tasks which will be deleted and which are considered as requirements in the standard.
4) Documenting and rationalising together all tailoring decisions, which consists only of one task, definable as the activity itself.

ISO/IEC 12207 includes an informative annex which provides guidance to its tailoring. Two tailoring levels are suggested: the first one, for adapting the standard to a specific business area (aviation, military, etc.); the second one, for each specific project or contract. We will talk about some issues of this annex when we explain the tailoring to the maintenance methodology.

B. The Maintenance process

Maintenance is considered by ISO/IEC 12207 as one of the primary processes of the software life cycle. It defines the activities and tasks which must be performed by the maintainer organization.

As all processes in this standard, Maintenance is made up of a set of activities and tasks. In the tenth task (third activity) of the Maintenance process, a Development process is introduced to carry-out the implementation of the modification. A strict follow-up of this Maintenance process would imply the execution of the thirteen activities of the Development process, with its corresponding set of tasks. Furthermore, this big number of works may imply overlapping of some activities and tasks (two acceptances of the customer, two system integrations, etc.).

III. USING THE TAILORING PROCESS TO BUILT-UP THE METHODOLOGY

This methodology spurts from a collaboration project between the University of Castilla-La Mancha, in Spain, and Atos ODS, a French multinational among whose main business activities is the outsourcing of software maintenance. This activity receives every day a growing importance on the software market [1], reason by which we think it is interesting to incorporate it to the methodology.

A direct and continuous contact between the university and the enterprise has characterised the building of MANTEMA: the first one, furnishing different, sequential and accumulative versions of the methodology; the second one, applying its own experience to refill lacks discovered and providing valuable suggestions.

A. Applying The Tailoring-Process Activities One By One

In this section we expose the sequential application of the Tailoring activities (see section II), to allow the reader the perception of different phases we have executed.

Activity 1. Identifying project environment.

Maintenance projects involve existing software systems. Normally, maintenance is an uncontrolled process, performed on demand, very often with very tight time margins to be carried-out.

Four types of maintenance are typically distinguished:

1) Corrective: when an error blocks the system and impedes its normal operation.
2) Perfective: when the addition of new functionalities is required.
3) Preventive: when some internal attributes of the software are going to be changed (maintainability, cyclomatic complexity, etc.), but with no change of neither the functionality nor the form of use of the system.
4) Adaptive: when the system is going to be adapted to a new operational environment.

We also divide the corrective into two subtypes:

1. Urgent corrective, when the error must be corrected very quickly.
2. Non-urgent corrective, when the error resolution can wait for some time.

In Figure 1 we summarize what parties of the environment affect every type of maintenance. Furthermore, every type has a different impact and cost, shown in Figure 2 [6].

It is interesting to consider the possibility of incorporating outsourcing activities and/or tasks to the maintenance process.

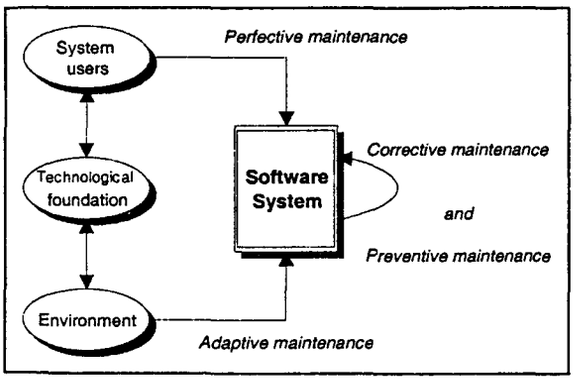

Figure 1. Sources of software maintenance.

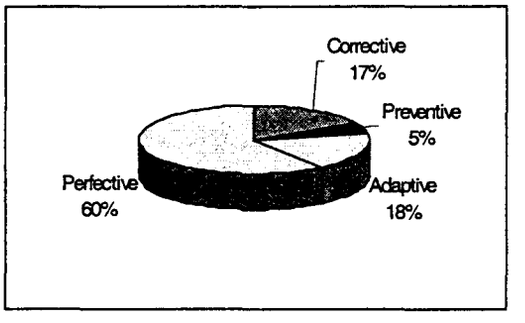

Figure 2. Costs of every maintenance type.

Activity 2. Soliciting inputs.

We have maintained interviews with all sort of Atos ODS personnel for nine months: programmers, heads of maintenance, project managers, etc. We have taken note of all their comments, and we have refined progressively all the different versions of the methodology.

Activity 3. Selecting processes, activities and tasks.

Our aim, when we begun the construction of this methodology, was to build a practical and quick guide for software maintenance, with support for metrics for this specific process. We do not desire to provide neither complicated crossed-references tables, nor different books to be capable of applying this methodology correctly. Then, we integrate into software maintenance some processes of ISO/IEC 12207, leaving some others (Infrastructure, Configuration Management, Improvement, Quality Assurance, Training and Operation) out of our bounds.

From the Maintenance process with the Development one expanded inside it, we can identify three groups of processes which served us as a foundation to build the methodology and that we show in Figure 3.

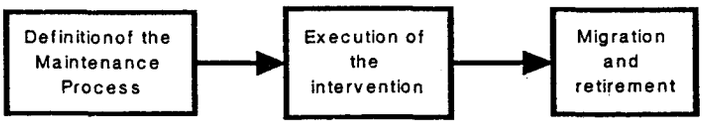

Figure 3. Foundation for the methodology, taken from ISO/IEC 12207.

However, bearing in mind among the interests of Atos ODS is the consideration of the outsourcing inside the maintenance, the incorporation of related sets of activities and tasks was needed. With the outsourcing, a customer organization gives to a provider organization the maintenance of a software subjected to a contract. Afterwards, the Definition of the Maintenance Process, the Execution of the Intervention and the Migration and Retirement are the customer's responsibility. In this way, we must add two nodes to Figure 3 graph: one at the beginning to carry-out the outsourcing contract, study the existing system that will be maintained, etc., and another to the end of the outsourcing relationship.

On the other hand, we saw previously that there are different forms of proceeding depending on the maintenance type, and we mentioned five (urgent-corrective, non-urgent corrective, perfective, preventive and adaptive). Therefore, we could split the central node of Figure 3 (Execution of the intervention) into five, one for every one of the five maintenance types. However, it is interesting to note that the practical application of the methodology in Atos ODS's customers showed that the latest four types of maintenance share common design lines and a very similar method of intervention. Then, one of the comments and suggestions mentioned in section A.2 (Soliciting inputs) by Atos ODS's directors was to group non-urgent corrective, perfective, preventive and adaptive maintenances into just one type: the predictable maintenance. To preserve terminology, we say "non-predictable maintenance" to mean urgent-corrective.

Then, with these two considerations, we redraw Figure 3 in this form:

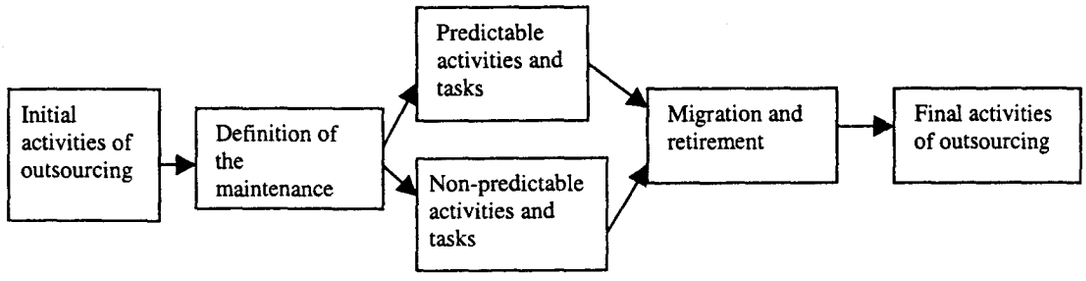

Figure 4. Definitive structure of the maintenance process.

Generally, activities contained in the two initial nodes and in the final one are carried out only once, although the second one, which is under the control of the provider organization (which will be the owner if there is not outsourcing) may be reexecuted so many times as necessary.

Below we list which are the activities and tasks which make up this methodology. However, we do not detail their contents, since they can be found in [8].

A.3.1. Initial activities of outsourcing.

Really, the initial node constitutes one special instance of a mixture of the ISO/IEC 12207 Acquisition and Supply processes, which have been integrated into this maintenance methodology. The following activity must be carried-out in the initial set of activities:

Activity 0. Initial study. This activity consists of the following tasks:

0.1 Beginning and information collection
0.2 Preparing the maintenance proposal
0.2 Contract

A.3.2. Definition of the Maintenance process.

Once signed the contract, the supplier of the maintenance service acquires the responsibility of the maintenance process. The first work to do is the second node of Figure 4: "Definition of the Maintenance Process". This set is related to the Process implementation activity of the ISO/IEC 12207 international standard. It is made up of one activity in this methodology. It is also a little special, since the maintainer must be acquainted with the information system and all their software products (because this knowledge was not revealed by the questionnaire, which is only a summary of the software characteristics).

The following activity must be performed in this set:

Activity 1. Process Implementation. This activity consists of the following tasks:

1.0 Knowledge acquisition
1.1 Developing plans
1.2 Defining Modification Request procedures
1.3 Implementing Configuration Management process
1.4 Preparing test environments

A.3.3. Execution of the intervention.

After task 1.4, the maintainer is prepared to carry-out the needed interventions of maintenance.

By reasons of space we will mention only the activities and tasks which compose the predictable (urgent-corrective) maintenance, which is the shortest.

Urgent-corrective interventions must be performed quickly, in very little time, against time to analyze alternatives, etc. Let us remember that, right now, ISO/IEC 12207 introduces a Development process inside the Maintenance. The Development process has many activities and tasks that we will not be able to use in urgent-corrective interventions. Then, and taking into account the singular character of software changes and modifications, we must disregard some activities of the Development process and limit the activities and tasks to develop to the following set:

Activity UC1. Error analysis. This activity is set up of the following task:

UC1.1 Investigating and analyzing causes

Activity UC2. Urgent-corrective intervention. This activity consists of the following tasks:

UC2.1 Making corrective actions

UC2.2 Filling in documentation

UC2.3 Verifying the modification

Activity UC3. Intervention closing. This activity consists of the following tasks:

UC3.1 Putting the software product to production environment.

A.3.4. Documenting tailoring decisions and rationale.

Possibly this paper is the required document for this activity.

A.3.5. Rest of the process and processes integration.

As we do not wish to expand this paper too much, we will say that the Migration and Retirement of the software are directly imported from ISO/IEC 12207. The last node of the graph we use to represent this maintenance process is made up of just one activity, which may be tailored depending on the conditions which have provoked the end of the outsourcing relationship, organizational procedures, etc.

We have mentioned that some processes of the International Standard are integrated into the maintenance process we define. As an example of the integration of the Documentation process (one of the Supporting processes of ISO/IEC 12207), the methodology provides templates for almost the possible documents generated during the process (Initial questionnaire, Maintenance proposal, Maintenance contract, Risk factor Table, Technical summary, Modification request, Performed corrective actions, Performed unitary tests, Diagnostic and possible solutions, Alternatives of implementation, Performed Perfective/Preventive/Corrective/Adaptive actions, List of software elements and properties to improve, Migration plan, Future migration notification, Product measures, Maintenance plan for a period, etc.).

Furthermore, in every task of the methodology, the document or set of documents that must be filled in are established.

IV. CONCLUSIONS AND FUTURE WORK.

In this work we have shown how we have used the Tailoring process of ISO/IEC 12207 to set up a methodology for supporting maintenance that -we believe- covers an existing gap in Software Engineering.

We have kept-up some original activities of the standard, but we have had to reject others. In the same manner, new activities and tasks have been added, in order to consider the outsourcing of software maintenance, as for the special treatment of the different types of maintenance.

Nowadays, we are in an advanced stage of the development of MANTOOL, an automatic tool which supports the methodology. We hope to present an evolved version of MANTOOL in the near future. It is hoped that this methodology will be used for MANTOOL's own maintenance.

V. REFERENCES

[1] Communications of the *ACM*. ACM, vol. 39, no 7, 1996. It contains an interesting monograph about outsourcing.

[2] Basili, V., Briand, L., Condon, S., Kim, Y., Melo, W. y Valett, J.D. *Understanding and predicting the Process Software Maintenance Releases*. Proceedings of the International Conference on Software Engineering, IEEE, 1996.

[3] Briand, L., Kim, Y., Melo, W., Seaman, C. and Basili, R. *Q-MOPP: Qualitative Evaluation of Maintenance Organizations, Processes and Products*. Journal of Software Maintenance, nº 10, pp. 249-278, 1998.

[4] Card, D.N. and Glass, R.L. *Measuring Software Design Quality*. Englewood Cliffs. USA, 1990.

[5] ISO/IEC 12207. *Information Technology. Software life cycle processes.*

[6] Piattini, M.G., Villalba, J., Ruiz, F., Fernández, I., Polo, M., Bastanchury, T. Y Martínez, M.A. *Mantenimiento del software. Conceptos, métodos, herramientas y outsourcing.* RA-MA, Madrid, Spain, 1998.

[7] Pigoski, T.M. *Practical Software Maintenance.* Wiley Computer Publishing. New York, USA, 1997.

[8] Polo, M., Piattini, M., Ruiz, F. y Calero, C. *MANTEMA: A complete rigorous methodology for supporting maintenance based on the ISO/IEC 12207 standard.* Proceedings of the 3rd Euromicro Conference on Software Maintenance and Reengineering. Amsterdam (The Netherlands), 1999.